

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

Requested Patent: JP2001344185A

Title:

INFORMATION PROCESSING DEVICE AND I/O EMULATION CONTROL METHOD

Abstracted Patent: JP2001344185 ;

Publication Date: 2001-12-14 ;

Inventor(s): ODA HIROYUKI ;

Applicant(s): TOSHIBA CORP ;

Application Number: JP20000163036 20000531 ;

Priority Number(s): ;

IPC Classification: G06F13/10 ;

Equivalents: ;

ABSTRACT:

PROBLEM TO BE SOLVED: To securely emulate the operation of an I/O device capable of dynamically mapping an I/O address with a software.SOLUTION: An I/O trap SMI is utilized as a program interface for starting an I/O emulation routine, and a software SMI to be periodically published is utilized as a program interface for starting an I/O address change detecting routine. The I/O address change detecting routine monitors a changeover of the I/O address of an I/O device 14 by an operating system, and when the I/O address is changed, the I/O address change detecting routine automatically renews a value of the I/O base address set in a trap address setting register 156 of an I/O trap SMI generating circuit 151 in response to the change of the I/O address.

(11)特許出願公開番号

特開2001-344185

(P2001-344185A)

(43)公開日 平成13年12月14日(2001.12.14)

(51) Int.Cl.⁷

識別記号

FI

 τ -7J-1° (参考)

G O 6 F 13/10

3 2 0

G O 6 F 13/10

320Z 5B014

審査請求 未請求 請求項の数 9 OL (全 9 頁)

(21)出願番号 特願2000-163036(P2000-163036)

(22) 出願日 平成12年5月31日(2000.5.31)

(71)出題人 000003078

株式会社東芝

東京都港区芝浦一丁目1番1号

(72)発明者 尾田 博幸

東京都青海市末広町2丁目9番地 株式会

社東芝青梅工場内

(74) 代理人 100058479

弁理士 鈴江 武彦 (外6名)

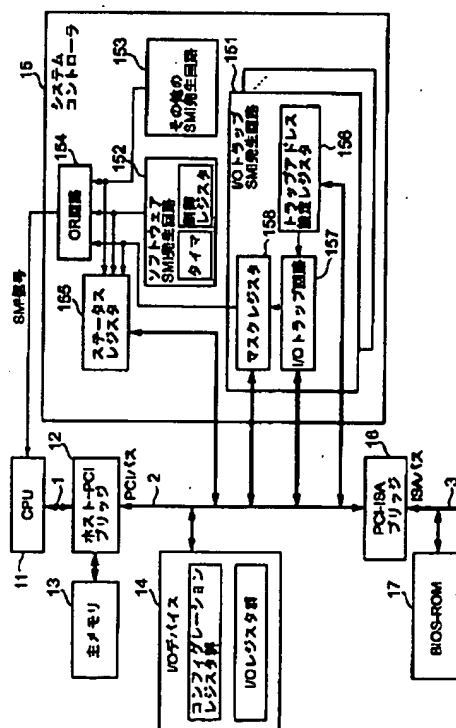
Fターム(参考) 5B014 FA11

(54) 【発明の名称】 情報処理装置およびI/Oエミュレーション制御方法

(57) 【要約】

【課題】I/Oアドレスを動的にマッピング可能なI/Oデバイスの動作をソフトウェアによって確実にエミュレーションする。

【解決手段】I/OトラップSMIはI/Oエミュレーションルーチンを起動するためのプログラム・インタフェースとして利用され、また定期的に発行されるソフトウェアSMIは、I/Oアドレス変更検出ルーチンを起動するためのプログラム・インタフェースとして利用される。I/Oアドレス変更検出ルーチンは、オペレーティングシステムによるI/Oデバイス14のI/Oアドレスの切り替えを監視し、変更されているならば、それに合わせてI/OトラップSMI発生回路151のトラップアドレス設定レジスタ156に設定されているI/Oベースアドレスの値を自動更新する。



【特許請求の範囲】

【請求項1】 I/Oデバイスに対してI/Oアドレスを動的にマッピング可能な情報処理装置であって、CPUと、

前記I/OデバイスのI/Oアドレスが割り込み起動用のI/Oアドレスとして設定され、前記割り込み起動用のI/Oアドレスを示す前記CPUからのI/Oアクセスをトラップして前記CPUに第1の割り込み信号を発行する手段と、

前記第1の割り込み信号によって起動され、前記I/Oアドレスで指定されたI/Oデバイスの動作をエミュレーションするI/Oエミュレーション手段と、

前記CPUによる前記I/Oデバイスに対するI/Oアドレスの設定変更を監視する手段と、

前記I/Oデバイスに対するI/Oアドレスの設定変更に応じて、前記割り込み起動用のI/Oアドレスの値を更新する手段とを具備することを特徴とする情報処理装置。

【請求項2】 前記I/Oアドレスの設定変更を監視する手段は、

前記I/Oデバイスのコンフィグレーション空間に設定されたI/Oベースアドレスの値を定期的にポーリングすることによって、前記I/Oデバイスに対するI/Oアドレスの設定変更の有無を検出することを特徴とする請求項1記載の情報処理装置。

【請求項3】 前記I/Oアドレスの設定変更を監視する手段は、

前記ポーリング時に、前記I/Oデバイスが動作禁止状態であるか否かを判別する手段と、

前記I/Oデバイスが動作禁止状態である場合、前記第1の割り込み信号の発行を禁止する手段とをさらに含むことを特徴とする請求項2記載の情報処理装置。

【請求項4】 前記I/Oアドレスの設定変更を監視する手段を、タイマによって定期的に発生される第2の割り込み信号によって起動させる手段をさらに具備することを特徴とする請求項1記載の情報処理装置。

【請求項5】 前記I/Oデバイスのコンフィグレーション空間に属する所定のアドレスが割り込み起動用の第2のアドレスとして設定され、前記割り込み起動用の第2のアドレスを示す前記CPUからのコンフィグレーションアクセスをトラップして前記CPUに第2の割り込み信号を発行する手段をさらに具備し、

前記I/Oアドレスの設定変更を監視する手段は、

前記第2の割り込み信号によって起動され、前記CPUからの前記コンフィグレーションアクセスが前記I/Oデバイスのコンフィグレーション空間に定義されるI/Oベースアドレスレジスタに対するライトであるか否かを判断し、その判断結果に基づいて前記I/Oデバイスに対するI/Oアドレスの設定変更の有無を検出することを特徴とする請求項1記載の情報処理装置。

【請求項6】 前記I/Oアドレスの設定変更を監視する手段は、

前記第2の割り込み信号による起動時に、前記I/Oデバイスが動作禁止状態であるか否かを判別する手段と、前記I/Oデバイスが動作禁止状態である場合、前記第1の割り込み信号の発行を禁止する手段とをさらに含むことを特徴とする請求項5記載の情報処理装置。

【請求項7】 エミュレーション対象のI/OデバイスのI/Oアドレスを示すCPUからのI/Oアクセスをトラップすることによって前記CPUに割り込み信号を発行することにより、前記I/Oアドレスで指定されたI/Oデバイスの動作をエミュレーションするためのプログラムを起動するI/Oエミュレーション制御方法であって、

前記CPUによる前記I/Oデバイスに対するI/Oアドレスの設定変更を監視するステップと、

前記I/Oデバイスに対するI/Oアドレスの設定変更に応じて、前記トラップ対象のI/Oアドレスの値を更新するステップとを具備することを特徴とするI/Oエミュレーション制御方法。

【請求項8】 I/Oアドレスの設定変更を監視するステップを、タイマによって定期的に発生される第2の割り込み信号によって起動させるステップをさらに具備することを特徴とする請求項7記載のI/Oエミュレーション制御方法。

【請求項9】 前記I/Oデバイスのコンフィグレーション空間に属する所定のアドレスを示す前記CPUからのコンフィグレーションアクセスをトラップして、前記I/Oアドレスの設定変更を監視するステップを起動させるための第2の割り込み信号を発生するステップをさらに具備することを特徴とする請求項7記載のI/Oエミュレーション制御方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明はI/Oデバイスの動作をソフトウェアによってエミュレーションすることが可能な情報処理装置およびI/Oエミュレーション制御方法に関する。

【0002】

【従来の技術】近年、コンピュータシステムの通信インタフェースの改良や、多機能化等に伴い、パーソナルコンピュータ等の情報処理装置においては、従来からのレガシーI/Oデバイスに加え、新たなI/Oデバイスの実装が要求され始めている。このような新たなI/Oデバイスとしては、例えばUSBインタフェースやIEEE1394インタフェース、あるいは携帯電話インタフェースなどのコントローラなどが挙げられる。

【0003】ところで、従来より、情報処理技術の分野においては、特定のハードウェアの機能をソフトウェアによってエミュレーションするというエミュレーション

技術が知られている。このエミュレーションは、例えば I/O デバイスの特定の I/O ポートに生じたバグを解消したり、あるいは実際には存在しない I/O ポートの機能を擬似的に提供するという用途に有効である。

【0004】

【発明が解決しようとする課題】しかし、最近の I/O デバイスは PC I 仕様のものが多く、適用するエミュレーション方法によっては正しいエミュレーション動作を実行できないなどの不具合が発生する危険があった。PC I 仕様の I/O デバイスは I/O アドレスを動的にマッピング可能に構成されており、オペレーティングシステムによって I/O アドレスマップの切り替えが行われることがあるからである。

【0005】本発明は上述の事情に鑑みてなされたものであり、I/O アドレスを動的にマッピング可能な I/O デバイスの動作を確実にエミュレーションすることが可能な情報処理装置および I/O エミュレーション制御方法を提供することを目的とする。

【0006】

【課題を解決するための手段】上述の課題を解決するため、本発明は、I/O デバイスに対して I/O アドレスを動的にマッピング可能な情報処理装置であって、CPU と、前記 I/O デバイスの I/O アドレスが割り込み起動用の I/O アドレスとして設定され、前記割り込み起動用の I/O アドレスを示す前記 CPU からの I/O アクセスをトラップして前記 CPU に第 1 の割り込み信号を発行する手段と、前記第 1 の割り込み信号によって起動され、前記 I/O アドレスで指定された I/O デバイスの動作をエミュレーションする I/O エミュレーション手段と、前記 CPU による前記 I/O デバイスに対する I/O アドレスの設定変更を監視する手段と、前記 I/O デバイスに対する I/O アドレスの設定変更に応じて、前記割り込み起動用の I/O アドレスの値を更新する手段とを具備することを特徴とする。

【0007】この情報処理装置においては、エミュレーション対象となる I/O デバイスの I/O アドレスが割り込み起動用の I/O アドレスとして設定されており、その割り込み起動用の I/O アドレスを示す I/O アクセスが CPU から発行されると、それがトラップされて CPU に第 1 の割り込み信号が発行される。この第 1 の割り込み信号により I/O エミュレーション手段が自動起動され、CPU からの I/O アドレスで指定された I/O デバイスの動作がエミュレーションされる。この方式においては、I/O エミュレーション手段を割り込みプログラムによって実現できるため、アプリケーションプログラムやドライバ等を変更することなく I/O の動作を容易にエミュレーションすることができるが、割り込み起動用の I/O アドレスはエミュレーション対象の I/O アドレスに一致させることが必要となる。

【0008】そこで、本発明では、エミュレーション対

象の I/O アドレスがオペレーティングシステムによって切り替えられても、それに追従して割り込み起動用の I/O アドレスを変更できるように、I/O アドレスの設定変更を監視する手段と、I/O デバイスに対する I/O アドレスの設定変更に応じて、割り込み起動用の I/O アドレスの値を更新する手段とが設けられている。これら手段より、割り込み起動用の I/O アドレスの値を、I/O デバイスに割り当てられる I/O アドレスに合わせることが可能となり、I/O デバイスの動作を確実にエミュレーションすることが可能となる。

【0009】前記 I/O アドレスの設定変更を監視する手段としては、I/O デバイスのコンフィグレーション空間に設定された I/O ベースアドレスの値を定期的にポーリングする手段を利用することができる。この場合、ポーリング時には、前記 I/O デバイスが動作禁止状態であるか否かを判別し、I/O デバイスが動作禁止状態である場合には第 1 の割り込み信号の発行を禁止することが好ましい。これにより、無駄な割り込みが発生するのを防止することができる。

【0010】また、前記第 2 の割り込み信号を、I/O デバイスのコンフィグレーション空間に属する所定のアドレスを示す CPU からのコンフィグレーションアクセスをトラップすることによって発生する仕組みを設け、これによって監視手段を起動することも可能である。これにより、ポーリングを利用する場合のようなタイマ割り込みなどの手法が不要となり、システム性能を高めることが可能となる。

【0011】

【発明の実施の形態】以下、図面を参照して本発明の実施形態を説明する。図 1 には、本発明の一実施形態に係わる情報処理装置の構成が示されている。この情報処理装置は例えばノートブックタイプのパーソナルコンピュータなどのコンピュータシステムであり、CPU 11、ホスト PC I ブリッジ 12、主メモリ 13、I/O デバイス 14、システムコントローラ 15、PCI-ISA ブリッジ 16、および BIOS-ROM 17 を内蔵している。

【0012】CPU 11 は本システム全体の動作制御およびデータ処理を実行する。この CPU 11 としては、システム管理割り込み SMI (SMI: System Management Interrupt) をサポートするものが利用される。即ち、CPU 11 はオペレーティングシステムやアプリケーション等を実行するための通常動作モードに加え、システム管理モード (SMM: System Management mode) と称されるシステム管理機能を実現するための動作モードを有している。

【0013】システム管理割り込み (SMI: System Management Interrupt) が CPU 11 に発行された時、CPU 11 の動作モードは、通常動作モード (リアルモード、プロテクトモード、または仮想 86 モード)

から、SMMにスイッチされる。SMMでは、I/Oデバイスの動作をエミュレーションするためのI/Oエミュレーションルーチンなどのシステム管理プログラムが実行される。

【0014】SMIはマスク不能割り込みNMIの一種であるが、通常のNMIやマスク可能割り込みINTRよりも優先度の高い、最優先度の割り込みである。このSMIを発行することによって、システム管理プログラムを、実行中のアプリケーションプログラムやオペレーティングシステム的环境に依存せずに起動することができる。

【0015】ホスト-PCIブリッジ12はCPUバス1とPCIバス2との間をつなぐブリッジLSIであり、ここには主メモリ13を制御するためのメモリコントローラも内蔵されている。I/Oデバイス14はPCI仕様のI/Oデバイス(PCIデバイス)であり、本システムにおいては、I/Oエミュレーションプログラムによるエミュレーション処理の対象となるターゲットデバイスとなる。このI/Oデバイス14は、そのI/Oアドレスを動的にマッピング可能に構成されている。I/Oデバイス14内には、コンフィグレーションレジスタ群およびI/Oレジスタ群が内蔵されている。I/Oレジスタ群に割り当てべきI/Oアドレス空間は、I/Oデバイス14内のコンフィグレーションレジスタ群内に定義されたI/Oベースアドレスレジスタの値(I/Oベースアドレス)に従って決定される。このI/Oベースアドレスの値は、オペレーティングシステムによって必要に応じて動的に書き換えられる。

【0016】システムコントローラ15には、CPU11へのSMI信号の発生を制御するためのハードウェアなどが組み込まれている。システムコントローラ15は実際には前述のホスト-PCIブリッジ12や後述のPCI-ISAブリッジ16に内蔵することができる。

【0017】PCI-ISAブリッジ16はPCIバス2とISAバス3間をつなぐブリッジLSIである。ISAバス3には、BIOS-ROM17が接続されている。BIOS-ROM17は、システムBIOS(Basic I/O System)を記憶するためのものであり、プログラム書き換えが可能ないようにフラッシュメモリによって構成されている。システムBIOSは、このシステム内の各種ハードウェアをアクセスするファンクション実行ルーチンを体系化したものであり、リアルモードで動作するように構成されている。このシステムBIOSには、システムのパワーオン時に実行されるIRTLルーチンと、各種ハードウェア制御のためのBIOSドライバ群が含まれている。各BIOSドライバは、ハードウェア制御のための複数の機能をオペレーティングシステムやアプリケーションプログラムに提供するためにそれら機能に対応する複数のファンクション実行ルーチン群を含んでいる。

【0018】また、BIOS-ROM17には、SMIサービスルーチンなどのSMMの中で実行されるシステム管理プログラム(SMM-BIOS)も格納されている。システム管理プログラムには、前述のI/Oエミュレーションルーチンや、オペレーティングシステムによるI/Oデバイス14のI/Oアドレスの切り替えを監視するためのI/Oアドレス変更検出ルーチンなどが含まれている。

【0019】次に、システムコントローラ15に設けられたSMI発生制御のためのハードウェア構成について説明する。

【0020】システムコントローラ15は、図示のように、I/OトラップSMI発生回路151、ソフトウェアSMI発生回路152、他の要因によるSMI発生回路153、OR回路154、およびSMIステータスレジスタ155を備えている。I/OトラップSMI発生回路151はI/OトラップSMI信号を発生させるための回路であり、予め決められたアドレス範囲に属するPCIバス2上のI/OアドレスをトラップしてI/OトラップSMI信号を発生する。このI/OトラップSMI発生回路151は、図示のように、トラップアドレス設定レジスタ156、I/Oトラップ回路157、およびマスクレジスタ158から構成されている。

【0021】トラップアドレス設定レジスタ156はCPU11がリード/ライト可能なレジスタであり、PCIバス2に接続されている。このトラップアドレス設定レジスタ156は、エミュレーション対象となるI/Oデバイス14のI/Oベースアドレスを割り込み起動用のI/Oアドレス(トラップアドレス)として保持する。このI/Oベースアドレスは、例えばシステムのパワーオン時にIRTLルーチンによってトラップアドレス設定レジスタ156に設定される。また、システム起動後にオペレーティングシステムによってI/Oデバイス14のI/Oベースアドレスが変更された場合には、それを検出したI/Oアドレス変更検出ルーチンによってトラップアドレス設定レジスタ156の内容がI/Oデバイス14のI/Oベースアドレスに合わせて書き換えられる。

【0022】I/Oトラップ回路157はI/Oリード/ライトサイクルにおいてPCIバス2上に出力されるI/Oアドレスとトラップアドレス設定レジスタ156のI/Oベースアドレスとを比較し、I/OアドレスがI/Oベースアドレスによって指定されるアドレス範囲に属する時にI/OトラップSMI信号を発生する。

【0023】マスクレジスタ158は、CPU11がリード/ライト可能なレジスタであり、PCIバス2に接続されている。このマスクレジスタ158は、I/OトラップSMI信号の発生を許可または禁止するためのマスクデータを保持する。

【0024】ソフトウェアSMI発生回路152はタイ

マおよび制御レジスタを有しており、制御レジスタで指定される所定のタイムアウト時間間隔でソフトウェアSMIを定期的に発生する。OR回路154は、I/OトラップSMI発生回路151、ソフトウェアSMI発生回路152、および他の要因によるSMI発生回路153のいずれかがSMI信号を発生した時、そのSMI信号をCPU11に供給する。SMIステータスレジスタ155は、SMI発生要因を示すステータスデータを保持するためのものであり、I/OトラップSMI発生回路151、ソフトウェアSMI発生回路152、および他の要因によるSMI発生回路153それぞれの出力を保持する。

【0025】図2には、I/OエミュレーションルーチンおよびI/Oアドレス変更検出ルーチンと、それらを起動するためのプログラム・インタフェースとの関係が示されている。図2に示されているように、I/OトラップSMIはI/Oエミュレーションルーチンを起動するためのプログラム・インタフェースとして利用され、またソフトウェアSMIは、I/Oアドレス変更検出ルーチンを起動するためのプログラム・インタフェースとして利用される。

【0026】次に、図3を参照して、本実施形態で用いられるI/Oエミュレーションの仕組みについて説明する。

【0027】オペレーティングシステムまたはアプリケーションプログラムがエミュレーション対象のI/Oアドレスを指定するINまたはOUT命令を実行すると、I/OトラップSMI発生回路151からI/OトラップSMIが発行され、これによってSMM-BIOSのI/Oエミュレーションルーチンが起動される。I/Oエミュレーションルーチンは、I/Oデバイス14の特定のI/Oポート（I/Oレジスタ）の機能をエミュレーションするものであり、最初にCPU11内のレジスタ値を参照してエミュレーション対象のI/Oポートを指定するINまたはOUT命令であることを確認する。そして、エミュレーション対象のI/Oポートを指定するINまたはOUT命令であった場合には、I/Oエミュレーションルーチンは、エミュレーション対象のI/Oポートの動作に対応する予め決められたエミュレーション処理を実行し、実際のハードウェアの代わりにオペレーティングシステムまたはアプリケーションプログラムに対してデータを返すなどの処理を行う。なお、エミュレーション対象のI/Oポートを指定するINまたはOUT命令に対してはハードウェア、つまりI/Oデバイス14は機能しない。

【0028】エミュレーション処理が完了すると、I/Oエミュレーションルーチンの割り込み元のオペレーティングシステムまたはアプリケーションプログラムに制御が戻される。

【0029】次に、図4のフローチャートを参照して、

I/Oアドレス変更検出ルーチンの処理について説明する。

【0030】前述したように、I/Oアドレス変更検出ルーチンはソフトウェアSMIによって定期的に起動される。I/Oアドレス変更検出ルーチンは、まず、エミュレーション対象となるI/Oデバイス14のコンフィグレーションレジスタをリードアクセスし、そこに設定されているI/Oデバイス14のI/Oベースアドレスの値をエミュレーション対象のI/Oアドレスとして取得する（ステップS101）。たとえI/Oベースアドレスの値が書き換えられていても、コンフィグレーションレジスタのコンフィグレーションアドレスは変わらないため、現在のI/OベースアドレスをI/Oデバイス14から正常に取得することができる。

【0031】次いで、I/Oアドレス変更検出ルーチンは、取得したI/Oベースアドレスが予め決められた無効アドレスであるか否かによって、I/Oデバイス14が動作禁止状態（ディスエーブル）に設定されているか、動作状態（イネーブル）に設定されているかを判断する（ステップS102）。取得したI/Oベースアドレスが無効アドレスである場合、つまりI/Oデバイス14が動作禁止状態である場合には（ステップS102のNO）、I/Oエミュレーションの必要が無いため、I/Oアドレス変更検出ルーチンは、I/OトラップSMIの発生を禁止するためのマスクデータをマスクレジスタ158に書き込み（ステップS114）、処理を終了する。これにより、不要なI/OトラップSMIの発生によってパフォーマンスが低下するのを防止することができる。

【0032】一方、取得したI/Oベースアドレスが有効アドレスである場合、つまりI/Oデバイス14が動作状態である場合には（ステップS102のYES）、I/Oアドレス変更検出ルーチンは、マスクレジスタ158のマスクデータを参照して、I/OトラップSMIの発生が許可されているか否かを判断する（ステップS104）。I/OトラップSMIの発生が禁止されている場合は、I/Oアドレス変更検出ルーチンは、I/OトラップSMIの発生を許可するためのマスクデータをマスクレジスタ158に書き込み、I/OトラップSMIを有効にする（ステップS105）。次に、I/Oアドレス変更検出ルーチンは、トラップアドレス設定レジスタ156からI/OトラップSMI起動用のI/Oアドレス（I/Oベースアドレス）を取得し（ステップS106）、ステップS101で取得したI/Oベースアドレスと比較する（ステップS107）。両者が一致しなかった場合には（ステップS107のNO）、I/Oアドレス変更検出ルーチンは、オペレーティングシステムがI/Oデバイス14のI/Oアドレスを動的に切り替えたものと判断して、ステップS101で取得したI/Oベースアドレスをトラップアドレス設定レジスタ1

56に書き込み、これによってI/OトラップSMI起動用のI/Oアドレスを現在のI/Oエミュレーションアドレスに変更する(ステップS108)。

【0033】このように、ソフトウェアSMIによってI/Oアドレス変更検出ルーチンを定期的に起動して、I/Oデバイス14のI/Oアドレスをポーリングすることにより、I/OトラップSMI起動用のI/OアドレスをI/Oエミュレーションアドレスの切り替えに追従させることが可能となる。これにより、オペレーティングシステムがI/Oデバイス14のI/Oアドレスを動的に切り替えても、その後もI/Oエミュレーションを正常に継続実行することが可能となる。

【0034】(第2実施形態)次に、本発明の第2実施形態について説明する。本第2実施形態は、I/Oアドレス変更検出ルーチンの起動方法およびI/Oアドレス変更検出ルーチンの処理手順のみが第1実施形態と異なっており、他の点は第1実施形態と同じである。

【0035】すなわち、本第2実施形態では、図5に示すように、I/OトラップSMI起動用I/Oアドレスの値が異なる2つのI/OトラップSMI発生回路151a、151bが用いられる。これらI/OトラップSMI発生回路151a、151bは共にシステムコントローラ15に内蔵されている。I/OトラップSMI発生回路151a、151bのどちらもその構成は図1のI/OトラップSMI発生151と同じである。I/OトラップSMI発生回路151aのトラップアドレス設定レジスタには、I/Oデバイス14のコンフィグレーション空間のベースアドレスがI/OトラップSMI起動用のコンフィグアドレスとして設定される。これにより、オペレーティングシステムがI/Oデバイス14のコンフィグレーションレジスタをアクセスしたときには、I/OトラップSMI発生回路151aから第1のI/OトラップSMI#1が発生される。また、I/OトラップSMI発生回路151bのトラップアドレス設定レジスタには、第1実施形態のI/OトラップSMI発生151と同様に、I/Oデバイス14のI/Oベースアドレスが設定される。これにより、オペレーティングシステムがI/Oデバイス14のエミュレーション対象のI/Oポートにアクセスするときには、I/OトラップSMI発生回路151bから第2のI/OトラップSMI#2が発生される。

【0036】図6には、I/OエミュレーションルーチンおよびI/Oアドレス変更検出ルーチンと、それらを起動するためのプログラム・インタフェースとの関係が示されている。図6に示されているように、第1のI/OトラップSMI#1はI/Oアドレス変更検出ルーチンを起動するためのプログラム・インタフェースとして利用され、また第2のI/OトラップSMI#2はI/Oシミュレーションルーチンを起動するためのプログラム・インタフェースとして利用される。

【0037】I/Oシミュレーションルーチンの処理手順については第1実施形態と同じであるので、以下、図7のフローチャートを参照して、本第2実施形態におけるI/Oアドレス変更検出ルーチンの処理手順について説明する。

【0038】オペレーティングシステムがI/Oデバイス14のコンフィグレーションレジスタをアクセスするための命令(コンフィグレーション空間へのOUT命令)を実行すると、I/OトラップSMI発生回路151aからI/OトラップSMI#1が発生され、これによってSMM-BIOSのI/Oエミュレーションルーチンが起動される。

【0039】I/Oアドレス変更検出ルーチンは、まず、CPU11内のレジスタを参照してコンフィグレジスタ内のI/Oベースアドレスレジスタ(I/Oベースレジスタ)へのライトであるか否かを判断する(ステップS111)。I/Oベースアドレスレジスタへのライトでは無い場合には(ステップS111のNO)、I/Oアドレス変更検出ルーチンはその時点で処理を終了する。

【0040】I/Oベースアドレスレジスタへのライトであった場合には(ステップS111のYES)、I/Oアドレス変更検出ルーチンは、I/Oデバイス14のコンフィグレーションレジスタをリードアクセスし、そこに設定されているI/Oデバイス14のI/Oベースアドレスの値をエミュレーション対象のI/Oアドレスとして取得する(ステップS112)。次いで、I/Oアドレス変更検出ルーチンは、取得したI/Oベースアドレスが予め決められた無効アドレスであるか否かによって、I/Oデバイス14が動作禁止状態(ディスエーブル)に設定されているか、動作状態(イネーブル)に設定されているかを判断する(ステップS113)。取得したI/Oベースアドレスが無効アドレスである場合、つまりI/Oデバイス14が動作禁止状態である場合には(ステップS113のNO)、I/Oエミュレーションの必要が無いため、I/Oアドレス変更検出ルーチンは、I/OトラップSMI#2の発生を禁止するためのマスクデータをI/OトラップSMI発生回路151bのマスクレジスタ158に書き込み(ステップS114)、処理を終了する。これにより、不要なI/OトラップSMIの発生によってパフォーマンスが低下するのを防止することができる。

【0041】一方、取得したI/Oベースアドレスが有効アドレスである場合、つまりI/Oデバイス14が動作状態である場合には(ステップS113のYES)、I/Oアドレス変更検出ルーチンは、I/OトラップSMI発生回路151bのマスクレジスタを参照して、I/OトラップSMI#2の発生が許可されているか否かを判断する(ステップS115)。I/OトラップSMI#2の発生が禁止されている場合は、I/Oアドレス

変更検出ルーチンは、I/OトラップSMI#2の発生を許可するためのマスクデータをI/OトラップSMI発生回路151bのマスクレジスタに書き込み、I/OトラップSMI#2を有効にする(ステップS116)。

【0042】次に、I/Oアドレス変更検出ルーチンは、I/OトラップSMI発生回路151bのトラップアドレス設定レジスタからI/OトラップSMI#2起動用のI/Oアドレス(I/Oベースアドレス)を取得し(ステップS117)、ステップS112で取得したI/Oベースアドレスと比較する(ステップS118)。両者が一致しなかった場合には(ステップS118のNO)、I/Oアドレス変更検出ルーチンは、オペレーティングシステムがI/Oデバイス14のI/Oアドレスを動的に切り替えたものと判断して、ステップS112で取得したI/OベースアドレスをI/OトラップSMI発生回路151bのトラップアドレス設定レジスタに書き込み、これによってI/OトラップSMI#2起動用のI/Oアドレスを現在のI/Oエミュレーションアドレスに変更する(ステップS119)。

【0043】このように、I/Oベースアドレスへの書き込みをトリガとしてI/Oアドレス変更検出ルーチンを起動して、I/OトラップSMI#2起動用のI/Oベースアドレスとの一致の有無をチェックすることにより、I/OトラップSMI#2起動用のI/OアドレスをI/Oエミュレーションアドレスの切り替えに追従させることが可能となる。これにより、オペレーティングシステムがI/Oデバイス14のI/Oアドレスを動的に切り替えても、その後もI/Oエミュレーションを正常に継続実行することが可能となる。

【0044】なお、上記第1および第2の各実施形態では、I/Oデバイス14のある特定のI/Oレジスタの動作をエミュレーションする場合を例示したが、アクセス先のレジスタを判別する仕組みを設けることにより、I/Oデバイス14の全I/Oレジスタの機能をエミュレーションすることも可能である。また、I/Oデバイス14として対象となるものには、内蔵PCIデバイスその他、PCIカードスロットに必要に応じて装着されるPCIカード型のデバイスも含まれる。

【0045】また、本発明は、上記各実施形態に限定されるものではなく、実施段階ではその要旨を逸脱しない範囲で種々に変形することが可能である。更に、上記実

施形態には種々の段階の発明が含まれており、開示される複数の構成要件における適宜な組み合わせにより種々の発明が抽出され得る。例えば、実施形態に示される全構成要件から幾つかの構成要件が削除されても、発明が解決しようとする課題の欄で述べた課題が解決でき、発明の効果の欄で述べられている効果が得られる場合には、この構成要件が削除された構成が発明として抽出され得る。

【0046】

【発明の効果】以上説明したように、本発明によれば、I/Oアドレスを動的にマッピング可能なI/Oデバイスの動作を確実にエミュレーションすることが可能となる。

【図面の簡単な説明】

【図1】本発明の第1実施形態に係るコンピュータシステムの構成を示すブロック図。

【図2】同第1実施形態におけるI/OエミュレーションルーチンおよびI/Oアドレス変更検出ルーチンとそれらを起動するためのプログラム・インタフェースとの関係を示す図。

【図3】同第1実施形態におけるI/Oエミュレーション処理を説明するためのフローチャート。

【図4】同第1実施形態におけるI/Oアドレス変更検出処理を説明するためのフローチャート。

【図5】本発明の第2実施形態で使用される2種類のI/OトラップSMI発生回路を説明するための図。

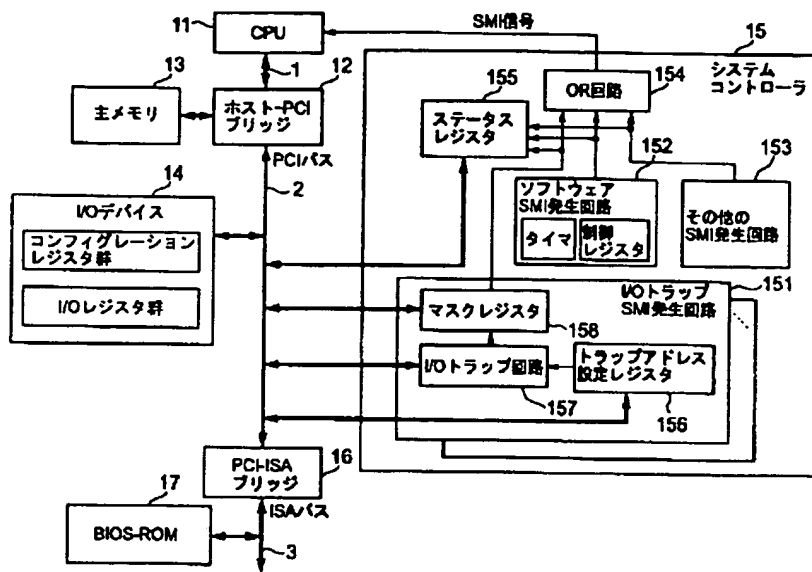
【図6】同第2実施形態におけるI/OエミュレーションルーチンおよびI/Oアドレス変更検出ルーチンとそれらを起動するためのプログラム・インタフェースとの関係を示す図。

【図7】同第2実施形態におけるI/Oアドレス変更検出処理を説明するためのフローチャート。

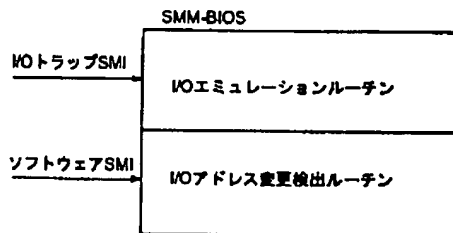
【符号の説明】

11…CPU
13…主メモリ
14…I/Oデバイス
15…システムコントローラ
17…BIOS-ROM
151, 151a, 151b…I/OトラップSMI発生回路
152…ソフトウェアSMI発生回路

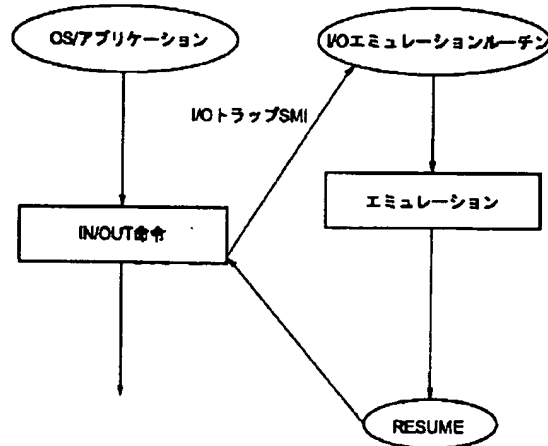
【図1】



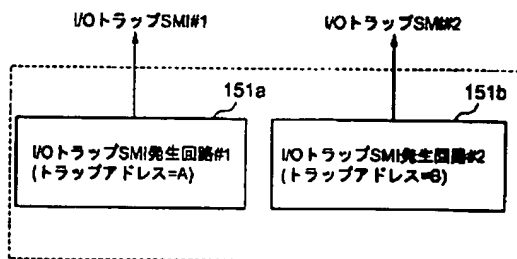
【図2】



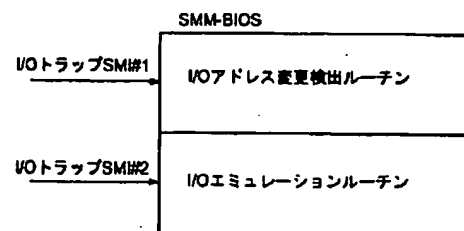
【図3】



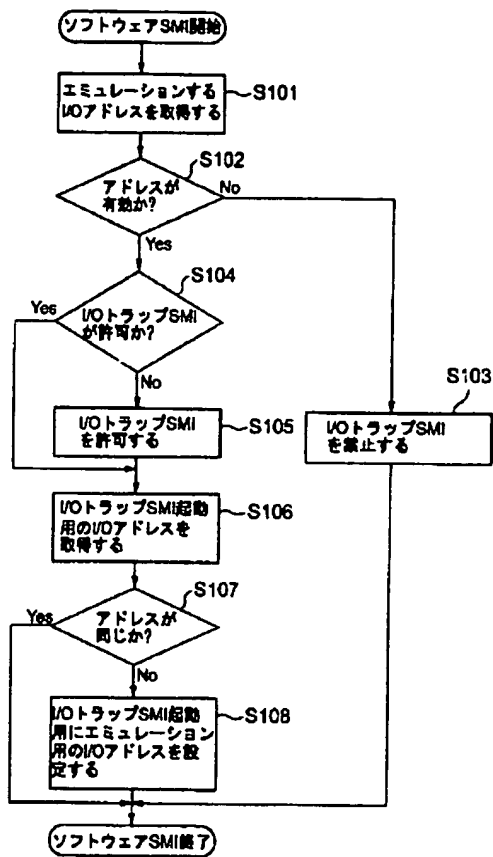
【図5】



【図6】



【図4】



【図7】

